Listing of Claims

This listing of claims replaces all prior versions, and listings, of claims in the application:

1. (Currently Amended) A data processing system[[,]] comprising:

a memory mapped non-volatile memory region ~~in~~ including one or more memory mapped executable instructions to initialize the data processing system;

a non-memory mapped non-volatile memory region, the non-memory mapped non-volatile memory region including one or more ~~additional~~ non-memory mapped executable instructions to initialize the data processing system; and

a third memory region in communication with at least one of the memory mapped non-volatile memory region and the non-memory mapped ~~static~~ non-volatile memory region.

2. (Currently Amended) The system of claim 1, wherein the memory mapped ~~static~~ non-volatile memory region comprises a first integrated circuit including a hardware mechanism to enable access to the one or more executable instructions using a processor memory read.

3. (Original) The system of claim 2, wherein the non-memory mapped non-volatile memory region comprises a second integrated circuit separate from the first integrated circuit.

4. (Currently Amended) The system of claim 3, wherein the second integrated circuit does not include a hardware mechanism to enable access to the one or more ~~additional~~ non-memory mapped executable instructions using a processor memory read.

5. (Original) The system of claim 1, wherein at least one of the memory mapped non-volatile memory region and the non memory-mapped non-volatile memory region is implemented as flash memory.

6. (Currently Amended) The system of claim 1, wherein the third memory region comprises cache memory.

7. (Original) The system of claim 6, further including a microprocessor, and wherein the cache memory is integrated with the microprocessor.

8. (Original) The system of claim 6, wherein the cache memory is partitioned to include a first region to store data and a second region to store one or more pages.

9. (Original) The system of claim 8, wherein the cache memory is partitioned to further include a third region to store one or more page tables.

10. (Currently Amended) A method of initializing a computer system, comprising:

(a) accessing memory mapped firmware using a processor memory read;

(b) executing an initialization instruction included in the memory mapped firmware;

(c) copying a page including ~~another~~ a second initialization instruction from non-memory mapped firmware to ~~another~~ a second memory region;

(d) translating an address in the second initialization instruction to a physical address of the copied page in the second memory region; and

~~(d)~~ (e) executing the ~~another~~ second initialization instruction.

11. (Currently Amended) The method of claim 10, wherein actions (a) through ~~(d)~~ (e) are performed prior to initializing a main memory of the computer system.

12. (Original) The method of claim 10, further comprising prior to (c), initializing at least a portion of a cache memory as random access memory.

13. (Currently Amended) The method of claim 10, further comprising prior to (c):

(b)(1) attempting to access a desired page of data stored in the ~~another~~ second memory region;

(b)(2) receiving a page fault signal indicating that the desired page of data is not stored in the ~~another~~ second memory region; and

(b)(3) copying the desired page of data from non-memory mapped firmware to the ~~another~~ second memory region.

14. (Currently Amended) The method of claim 13, further comprising receiving a fault serviced signal indicating that the desired of page of data is in the ~~another~~ second memory region.

15. (Currently Amended) The method of claim 14, further comprising executing an instruction included in the desired page of data stored in the ~~another~~ second memory region.

16. (Currently Amended) A chipset[[,]] comprising:

a memory mapped non-volatile memory region in including one or more memory mapped executable instructions to initialize the data processing system;

a non-memory mapped non-volatile memory region, the non-memory mapped non-volatile memory region including one or more ~~additional~~ non-memory mapped executable instructions to initialize the data processing system; and

a third memory region in communication with at least one of the memory mapped non-volatile memory region and the non-memory mapped ~~static~~ non-volatile memory region.

17. (Currently Amended) The chipset of claim 16, wherein the memory mapped ~~static~~ non-volatile memory region comprises a first integrated circuit included in the chipset, the first integrated circuit including a hardware mechanism to enable access to the one or more executable instructions using a processor memory read.

18. (Original) The chipset of claim 17, wherein the non-memory mapped non-volatile memory region comprises a second integrated circuit included in the chipset, the second integrated circuit separate from the first integrated circuit.

19. (Currently Amended) The chipset of claim 18, wherein the second integrated circuit does not include a hardware mechanism to enable access to the one or more ~~additional~~ non-memory mapped executable instructions using a processor memory read.

20. (Original) The chipset of claim 18, wherein at least one of the memory mapped non-volatile memory region and the non memory-mapped non-volatile memory region is implemented as flash memory.

21. (Original) The chipset of claim 16, wherein the chipset is in communication with a microprocessor.

22. (Currently Amended) The chipset of claim 16, wherein the third memory region comprises cache memory.

23. (Original) The chipset of claim 22, wherein the cache memory is configurable to be partitioned to include a first region to store data and a second region to store one or more pages.

24. (Original) The chipset of claim 23, wherein the cache memory is configurable to be partitioned to further include a third region to store one or more page tables.

25. (Currently Amended) An article of manufacture comprising a machine accessible medium containing code having instructions that, when executed, cause the machine to:

(a) access memory mapped firmware using a processor memory read;

(b) execute an initialization instruction included in the memory mapped firmware;

(c) copy a page including ~~another~~ a second initialization instruction from non-memory mapped firmware to ~~another~~ a second memory region;

(d) translating an address in the second initialization instruction to a physical address of the copied page in the second memory region; and

~~(d)~~ (e) execute the another instruction.

26. (Currently Amended) The article of claim 25, wherein the instructions cause the machine to perform (a) through ~~(d)~~ (e) prior to initializing a main memory of the computer system.

27. (Original) The article of claim 25, the instructions further causing the machine to initialize at least a portion of a cache memory as random access memory prior to (c).

28. (Currently Amended) The article of claim 25, the instructions further causing the machine to, prior to (c):

(b)(1) attempt to access a desired page of data stored in the ~~another~~ second memory region; and

(b)(2) upon receiving a page fault signal indicating that the desired page of data is not stored in the ~~another~~ second memory region, copy the desired page of data from non-memory mapped firmware to the ~~another~~ second memory region.

29. (Currently Amended) The article of claim 28, the instructions further causing the machine to receive a fault serviced signal indicating that the desired of page of data is in the ~~another~~ second memory region.

30. (Currently Amended) The article of claim 29, the instructions further causing the machine to execute an instruction included in the desired page of data stored in the ~~another~~ second memory region.

31. (New) The data processing system of claim 1, further comprising a paging handler to access the non-memory mapped non-volatile memory region and load the non-memory mapped executable instructions into the memory region.

32. (New) The method of claim 10, further comprising partitioning the second memory region to include a region for data and a region for pages.

33. (New) The method of claim 32, further comprising partitioning the second memory region to include a region for page tables.

34. (New) The chipset of claim 16, further comprising a paging handler to implement a paging mechanism.

35. (New) The article of claim 25, wherein the instructions cause the machine to partition the second memory region to include a region for data and a region for pages.

36. (New) The article of claim 25, wherein the instructions cause the machine to partition the second memory region to include a region for page tables.